

## CLAIMS

What is claimed is:

1. a)

A computer-implemented method of conducting a consecutive betting process for investors, the computer having a betting exchange unit for performing the following steps:

- a) identifying an uncertain event having potential outcomes  $O_1, \dots O_m$ , where  $m \geq 2$ ;
- b) initializing a first betting cycle;
- c) receiving bets  $B_1, \dots, B_m$  from the investors for each of the potential outcomes  $O_1, \dots O_m$  during the first betting cycle to accumulate an initial bet total  $B_{tot}$ ;
- d) issuing equal numbers  $OS(1), \dots OS(m)$  of outcome shares such that  $OS(1) = \dots = OS(m)$ , the outcome shares corresponding to the potential outcomes  $O_1, \dots O_m$ ;
- e) assigning a share value  $SV$  to each of the outcome shares;
- e) assigning quote values  $Q_1, \dots, Q_m$  to each of the outcome shares such that  $Q_1 = (SV * B_1) / B_{tot}, \dots, Q_m = (SV * B_m) / B_{tot}$ ; and
- f) distributing the outcome shares to the investors.

2. The method of claim 1, further comprising the steps of:

- g) monitoring an actual outcome  $OA$  of the future event; and
- h) selecting from among the outcome shares winning shares  $WS$  corresponding to the actual outcome  $OA$  and determining a number of winning shares  $NWS$ .

3. The method of claim 2, wherein the number of winning shares  $NWS$  is selected such that  $NWS * SV = B_{tot}$ .

5. The method of claim 2, wherein the step of monitoring the actual outcome OA is performed by a data acquisition unit.
6. The method of claim 1, wherein the investors comprise real investors and artificial investors.
7. The method of claim 6, wherein at least one artificial betting entity places a minimum initial bet  $B_{\min}$  on any of the potential outcomes  $O_1, \dots, O_m$  for which corresponding initial bets  $B_1, \dots, B_m$  are zero.
8. The method of claim 6, wherein the real investors are connected to the betting exchange unit by a communication network.
9. The method of claim 1, the method further comprising the following steps:
- i) initializing a subsequent betting cycle;
  - j) receiving amounts of money  $IM(1), \dots, IM(m)$  corresponding to subsequent bets  $B_1, \dots, B_m$  from the investors on each of the potential outcomes  $O_1, \dots, O_m$  during the subsequent betting cycle;
  - k) receiving numbers  $IS(1), \dots, IS(m)$  of incoming shares in outcomes  $O_1, \dots, O_m$  from the investors during the subsequent betting cycle; and
  - l) re-assigning the quote values  $Q_1, \dots, Q_m$  to preserve an equal number of outstanding shares in outcomes  $O_1, \dots, O_m$  such that  $OS(1) - IS(1) = \dots = OS(m) - IS(m)$ , wherein  $OS(i)$  are numbers of outcome shares for outcomes  $O_1 \dots O_m$  newly issued during the subsequent betting cycle.
- ).

5h  
a1

10. The method of claim 9, wherein the numbers of incoming outcome shares and newly issued outcome shares exchanged are in accordance with the reassigned quote values  $Q_1, \dots, Q_m$ .

11. The method of claim 9, further comprising the steps of:

- m) monitoring an actual outcome OA of the future event; and
- n) selecting from among the outcome shares winning shares WS corresponding to the actual outcome OA and assigning a normalized share value SV to each of the winning shares WS.

12. The method of claim 11, wherein the normalized share value SV is selected such that  $NWS \cdot SV = B_{tot}$ , where NWS is the number of winning shares.

13. The method of claim 11 wherein said normalized share value SV is equal to a unit of currency.

14. The method of claim 9, further comprising:

- g) determining amounts of outgoing money  $OM(1), \dots, OM(m)$  for each kind of outcome share, wherein each amount of outgoing money  $OM(i)$  is determined by  $OM(i) = \frac{IM(i) \cdot IS(i)}{OS(i)}$ .

15. The method of claim 14, wherein the revised quotes  $Q_1 \dots Q_m$  are determined by  $Q_i = \frac{IM(i)}{OS(i)} = \frac{OM(i)}{IS(i)}$ .

16. The method of claim 9 wherein step d) includes solving a polynomial of having  $m+1$  roots.

17. A system for conducting a consecutive betting process for investors placing bets  $B_1, \dots, B_m$  on potential outcomes  $O_1, \dots, O_m$  of a future event, where  $m \geq 2$ , the system having:

- a) a means for sending the bets  $B_1, \dots, B_m$  from the investors;
- b) a betting exchange unit for initiating a first betting cycle and receiving the bets  $B_1, \dots, B_m$  from the investors during the first betting cycle, the bets  $B_1, \dots, B_m$  accumulating to an initial bet total  $B_{tot}$ , the betting exchange unit further comprising:
  - i) a computing unit for issuing equal numbers  $OS(1), \dots, OS(m)$  of outcome shares such that  $OS(1) = \dots = OS(m)$ , the outcome shares corresponding to the potential outcomes  $O_1, \dots, O_m$ , the computing unit assigning a share value  $SV$  to each of the outcome shares, the computing unit further assigning quote values  $Q_1, \dots, Q_m$  to each of the outcome shares  $OS(1), \dots, OS(m)$  such that  $Q_1 = (SV * B_1) / B_{tot}, \dots, Q_m = (SV * B_m) / B_{tot}$ ; and
  - ii) a distributing unit for distributing the outcome shares to the investors.

18. The system of claim 17, wherein the computing unit further comprises an interface for receiving an actual outcome  $OA$  of the future event, the computing unit selecting from among the outcome shares winning shares  $WS$  corresponding to the actual outcome  $OA$  and assigning a normalized share value  $SV$  to each of the winning shares  $WS$ .

19. The system of claim 18, further comprising a data acquisition unit for monitoring the actual outcome OA, the data acquisition unit being connected to the interface.

20. The system of claim 17, wherein the investors comprise real investors and artificial investors.

21. The system of claim 17, wherein the means for sending the bets  $B_1, \dots, B_m$  comprises a communication network.

22. The system of claim 17, wherein the betting exchange unit is programmed to initialize a subsequent betting cycle for receiving subsequent bets  $B_1, \dots, B_m$  from the investors on each of the potential outcomes  $O_1, \dots, O_m$  during the subsequent betting cycle and for receiving shares  $IS(1), \dots, IS(m)$  from the investors during the subsequent betting cycle, and the computing unit is programmed to re-assign the quote values  $Q_1, \dots, Q_m$  to preserve an equal number of outstanding shares in outcomes  $O_1, \dots, O_m$  such that

$$OS(1) - IS(1) = \dots = OS(m) - IS(m).$$

23. The system of claim 22, wherein the computing unit further comprises an interface for receiving an actual outcome OA of the future event, the computing unit selecting from among the outcome shares winning shares WS corresponding to the actual outcome OA and assigning a normalized share value SV to each of the winning shares WS.

24. The system of claim 23, further comprising a data acquisition unit for monitoring the

DSK-101

3  
4

~~actual outcome OA, the data acquisition unit  
being connected to the interface~~

actual outcome OA, the data acquisition unit  
being connected to the interface

## APPENDIX

5 BETTING EXCHANGE MACHINE CODE (PowerBasic 3.5)

©1995 Dimitri P. M. Speck

```

DEFQUD N,I,G                                'long
DEFEXT E,K,X,S                              'exact
DIM IM(100),IS(100), KiZ(100),N(100),Kurs(100) 'up to 100
10                                           'events
                                           dimensioned

```

## 'INPUT

```

Z=5      'Number of events
15 IM(1)=1500    'IncomingMoney: Sum, that is invented during
IM(2)=20000    'one period for event 1,2,3...
IM(3)=25000
IM(4)=50000    'IM() should be always >0 in this code
IM(5)=80000    '(or else many exceptions would be necessary)
20 IS(1)=1000   'IncomingShares: Number of all shares, which
IS(2)=80000    'withdrawing participants return to the machine
IS(3)=10000    'during 1 period for event 1,2,3...
IS(4)=20000
25 IS(5)=45000

```

## 'TECHNICAL PARAMETERS

```

S=10000      'Factor of startsteps
Weight=.2    'Weight for selection-process. The selection-
30           'process selects automatically an event (what
           'in the script is specified as event 1)
Force=0      '0=automatic selection
           'other numbers=manual selection (ignores
           'automatic selection process)

```

DSK-101

```

35  Srd=4          'Stepreducing each change of direction
Exact=0.000000001 'Exactness

'Script:          IM(1)*IS(1)    IM(2)*IS(2)
40  'IS(1)+IM(1)+...- ----- - ---- ----- - ...- OS(1)=0
    '              OS(1)    OS(1)-IS(1)+IS(2)

'Code             KiZ(Choice)      KiZ()
'   Sum&&         - ----- - ---- - ...- X = Ergn
45  '             X+N(Choice)      X+N()

Choice=0:IS(0)=10^18:FOR i=1 TO Z          'SELECTION PROCESS
    IF IM(i)-IS(i)*Weight>IM(choice)-IS(Choice)*Weight
    THEN Choice=i
50 NEXT:IF NOT Force=0 THEN Choice=Force

    'PUTTING TOGETHER THE CONSTANTS OF THE BETTING EXCHANGE
    'MACHINE POLYNOMIAL

55  Sum&&=IS(Choice)          'Script: IS(1)
FOR i=1 TO Z:Sum&&=Sum&&+IM(I):NEXT        'Script: IS(1)+
                                           'IM(1)+IM(2)+IM(3)+...
FOR i=1 TO Z:KiZ(I)=IM(I)*IS(I):NEXT      'Script: IM(i)*IS(i)
                                           '(numerator)
60  FOR i=1 TO Z:N(I)=-IS(Choice)+IS(I):NEXT 'Script:-IS(1)+IS(i)
                                           '(part of the
                                           'denominator)

ON ERROR GOTO Zerodivision
65  Stepiz=IM(Choice)/s          'Startvalue of the stepsize
X=IM(Choice)-(4*stepsiz)        'Startvalue of X

```



```

70  'FINDING THE SOLUTION OF THE BETTING EXCHANGE MACHINE
    'POLYNOMIAL
Restart:
                                     'Initialization
FOR i=1 TO Z:Ergn=Ergn-KiZ(i)/(X+N(i)):NEXT:Ergn=Ergn+Sum&&-X
75
Verf:                               'Loop to right side (up)
DO                                  'tries to find zero
    Stp=stepsiz
    X=X+Stp:Ergv=Ergn:Ergn=0
80  FOR i=1 TO Z:Ergn=Ergn-KiZ(i)/(X+N(i)):NEXT:Ergn=Ergn+Sum&&-X
    IF Ergv=Ergn THEN GOTO Outputting 'if identical (zero or
                                     'extremum)

    IF ign=0 THEN
        IF NOT SGN(Ergv)=SGN(Ergn) THEN EXIT LOOP 'changed sign?
85  END IF
    IF Ergn>Ergv THEN rtg=1 ELSE rtg=2 '1=direction up, 2=down
    IF rtg+rtgv=3 THEN EXIT LOOP      'Changed direction?
                                     '(extremum at zero?)

    rtgv=rtg
90  IF ign=1 Then rtg=0:rtgv=0:ign=0  'Don't ignore
LOOP
Stepsiz=stepsiz/Srd:rtgv=0          'step smaller

DO                                  'Loop to left side (down)
95  Stp=Stepsiz
    X=X-Stp:Ergv=Ergn:Ergn=0
    FOR i=1 TO Z:Ergn=Ergn-KiZ(i)/(X+N(i)):NEXT:Ergn=Ergn+Sum&&-X
    IF Ergv=Ergn THEN GOTO Outputting
    IF NOT SGN(Ergv)=SGN(Ergn) THEN EXIT LOOP
100 IF Ergn > Ergv THEN rtg=2 ELSE rtg=1
    IF rtgv+rtg=3 THEN EXIT LOOP
    rtgv=rtg

```

DSK-101

LOOP

105 Stepsiz=Stepsiz/Srd:rtgv=0

GOTO Verf

Outputting:

'QUOTES

110 FOR i=1 TO Z:Kurs(i)=IM(i)/(X-IS(Choice)+IS(i)):next

Sumk=0

'WRONG ROOT?

FOR i=1 TO Z:Sumk=Kurs(i)+Sumk:NEXT

'Sum of all quotes

ign=1:rtg=0:rtgv=0:Stepsiz=IM(Choice)/s

115 IF ABS(Sumk-1)>Exact THEN GOTO Verf

'Sum is not equal 1:

'try next root

FOR i=1 TO Z

IF Kurs(i)<(0-Exact) THEN GOTO verf

'Negative quote: try

'next root

120 NEXT

'OUTPUT (quotes)

FOR i=1 TO Z:PRINT "Kurs(";I;")=";USING"#.#####";

Kurs(i):NEXT:END

125 zerodivision:

'Handling of 'division by zero'

X=X+Exact:Stepsiz=IM(Choice)/s:Ergn=0

'Restart further

'right

rtg=0:rtgv=0:ign=0

RESUME RESTART

130